Exhaustive search of optimal formulae for bilinear maps

Svyatoslav Covanov
Supervisors: Jérémie Detrey and Emmanuel Thomé

Team CARAMBA

January 16, 2017

Karatsuba

How to multiply two polynomials $A = a_0 + a_1 X$ and $B = b_0 + b_1 X$?

Karatsuba

How to multiply two polynomials $A = a_0 + a_1 X$ and $B = b_0 + b_1 X$?

$A \cdot B = a_0 b_0 + (a_1 b_0 + a_0 b_1)X + a_1 b_1 X^2$

1. **Naive multiplication:**
   - $\pi_0 = \boldsymbol{a_0 b_0}$, $\pi_1 = \boldsymbol{a_1 b_0}$, $\pi_2 = \boldsymbol{a_0 b_1}$ and $\pi_3 = \boldsymbol{a_1 b_1}$.
   - We have $A \cdot B = \pi_0 + (\pi_1 + \pi_2)X + \pi_3 X^2$.
2. **Karatsuba**:
   - $\pi_0 = \boldsymbol{a_0 b_0}$, $\pi_1 = \boldsymbol{a_1 b_1}$ and $\pi_2 = \boldsymbol{(a_0 + a_1)(b_0 + b_1)}$.
   - We have $A \cdot B = \pi_0 + (\pi_2 - \pi_0 - \pi_1)X + \pi_1 X^2$.

   The bilinear rank is smaller than 3.

Short product

$$\Pi_\ell : \begin{array}{ccc} K[X]_{<\ell} \times K[X]_{<\ell} & \to & K[X]_{<\ell} \\ (A, B) & \mapsto & A \cdot B \bmod X^\ell \end{array}$$

For $\ell = 3$,

$$\Pi_3 : \left(\begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix}, \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} \right) \mapsto \begin{pmatrix} a_0 b_0 \\ a_1 b_0 + a_0 b_1 \\ a_2 b_0 + a_1 b_1 + a_0 b_2 \end{pmatrix} = \begin{pmatrix} \pi_0 \\ \pi_1 \\ \pi_2 \end{pmatrix}$$

Optimal decomposition: $\mathrm{rk}(\Pi_3) = 5$

$$\boldsymbol{a_0 b_0,\ a_1 b_1,\ a_2 b_2,\ (a_0 + a_1)(b_0 + b_1),\ (a_0 + a_2)(b_0 + b_2)}$$

Matrix formalism

$$\pi_0 = \begin{pmatrix} a_0 & a_1 & a_2 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} = a_0 b_0$$

$$\pi_1 = \begin{pmatrix} a_0 & a_1 & a_2 \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} = a_1 b_0 + a_0 b_1$$

$$\pi_2 = \begin{pmatrix} a_0 & a_1 & a_2 \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} = a_2 b_0 + a_1 b_1 + a_0 b_2$$

Matrix representation of formulae:

$$\underbrace{\begin{pmatrix}1&0&0\\0&0&0\\0&0&0\end{pmatrix}}_{a_0b_0},\ \underbrace{\begin{pmatrix}0&0&0\\0&1&0\\0&0&0\end{pmatrix}}_{a_1b_1}\ \underbrace{\begin{pmatrix}0&0&0\\0&0&0\\0&0&1\end{pmatrix}}_{a_2b_2}\ \underbrace{\begin{pmatrix}1&1&0\\1&1&0\\0&0&0\end{pmatrix}}_{(a_0+a_1)(b_0+b_1)}\ \underbrace{\begin{pmatrix}1&0&1\\0&0&0\\1&0&1\end{pmatrix}}_{(a_0+a_2)(b_0+b_2)}$$

Decomposition:

$$\underbrace{\begin{pmatrix}0&1&0\\1&0&0\\0&0&0\end{pmatrix}}_{\pi_1} = \begin{pmatrix}1&1&0\\1&1&0\\0&0&0\end{pmatrix} - \begin{pmatrix}1&0&0\\0&0&0\\0&0&0\end{pmatrix} - \begin{pmatrix}0&0&0\\0&1&0\\0&0&0\end{pmatrix}$$

$$\underbrace{\begin{pmatrix}0&0&1\\0&1&0\\1&0&0\end{pmatrix}}_{\pi_2} = \begin{pmatrix}0&0&0\\0&1&0\\0&0&0\end{pmatrix} + \begin{pmatrix}1&0&1\\0&0&0\\1&0&1\end{pmatrix} - \begin{pmatrix}1&0&0\\0&0&0\\0&0&0\end{pmatrix} - \begin{pmatrix}0&0&0\\0&0&0\\0&0&1\end{pmatrix}$$

Generators definition

Let $\mathcal{G}$ be the set of generators

$$\mathcal{G} = \left\{ (A, B) \mapsto (\sum_i \lambda_i a_i)(\sum_j \mu_j b_j) \mid \lambda_i \in K, \mu_j \in K \right\}.$$

For the short product Π_3 over $\mathbb{F}_2$, the generators are

$$\begin{Bmatrix} a_0 b_0, & a_1 b_0, & a_2 b_0, & a_0 b_1, & a_1 b_1, & a_2 b_1, \\ a_0 b_2, & a_1 b_2, & a_2 b_2, & (a_0 + a_1) b_0, & (a_0 + a_2) b_0, & \dots \end{Bmatrix}$$

$$\downarrow$$

$$\left\{ \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \dots \right\}$$

Problem to be solved

Let $T_\ell =$

$$\operatorname{Span}\left(\begin{pmatrix}1 & 0 & \cdots & 0\\ 0 & 0 & \iddots & 0\\ \vdots & \iddots & \iddots & \vdots\\ 0 & 0 & \cdots & 0\end{pmatrix}, \begin{pmatrix}0 & 1 & \cdots & 0\\ 1 & 0 & \iddots & 0\\ \vdots & \iddots & \iddots & \vdots\\ 0 & 0 & \cdots & 0\end{pmatrix}, \ldots, \begin{pmatrix}0 & 0 & \cdots & 1\\ 0 & 0 & \iddots & \vdots\\ \vdots & \iddots & \iddots & 0\\ 1 & 0 & \cdots & 0\end{pmatrix}\right).$$

Problem to be solved:

Find all free families of minimal size $\mathcal{F} \subset \mathcal{G}$ satisfying $T_\ell \subset \operatorname{Span}(\mathcal{F})$.

Definition

Let $r \geq 0$. We denote by $\mathscr{S}_r$ all subspaces $V \subset \mathcal{M}_{\ell,\ell}$ such that there exists $\{g_0, \ldots, g_{r-1}\}$ a free family of $\mathcal{G}$ satisfying

$$V = \operatorname{Span}(g_0, \ldots, g_{r-1}).$$

We denote by $\mathscr{S}_{r,T}$ all subspaces $V \in \mathscr{S}_r$ such that $T \subset V$.

Naive algorithm

Naive algorithm

Input: ℓ, r
Output: $\mathscr{S}_{T_\ell, r}$

$\mathcal{S} \leftarrow \emptyset$
for $V \in \mathscr{S}_r$ **do** $\triangleright\ \mathscr{S}_r = \{\mathrm{Span}(g_0, \ldots, g_{r-1}) \mid \forall i,\ g_i \in \mathcal{G}\}$
  if $T_\ell \subset V$ **then**
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{V\}$
  end if
end for
return $\mathcal{S}$

Complexity: $\#\mathscr{S}_r \leq \binom{\#\mathcal{G}}{r}$. For $\ell = 3$ and $K = \mathbb{F}_2$, we have

$\#\mathscr{S}_5 = 157,535 \ll 1,906,884 = \binom{49}{5}$.

Incomplete basis improvement

BDEZ '12 (Barbulescu, Detrey, Estibals, Zimmermann)

Input: ℓ, r
Output: $\mathscr{S}_{T_\ell,r}$

```
S ← ∅
for W ∈ 𝒮_{r−ℓ} do
    if T_ℓ + W ∈ 𝒮_r then
        S ← S ∪ {T_ℓ + W}
    end if
end for
return S
```

Complexity: $\#\mathscr{S}_{r-\ell} \le \binom{\#\mathcal{G}}{r-\ell}$. For $\ell = 3$, $\#\mathscr{S}_2 = 980 \ll 157,535$.

Automorphisms

We consider the action of couples (P, Q) (P and Q in GL_ℓ) on $M \in \mathcal{M}_{\ell,\ell}$:

$$(P, Q) \cdot M = P \cdot M \cdot Q^T.$$

Let $\mathrm{Stab}(T_\ell)$ be the group of (P, Q) such that

$$\forall M \in T_\ell,\ (P, Q) \cdot M \in T_\ell.$$

Example

For $\ell = 3$ and $P = Q = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$,

$$(P, Q) \cdot \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} = \pi_1 + \pi_2 \in T_3.$$

Search with stabilizers

Input: ℓ, r

Output: $\mathscr{S}_{T_\ell, r}$

$\mathcal{S} \leftarrow \emptyset$

for $W \in \mathscr{S}_{r-\ell} / \mathrm{Stab}(T_\ell)$ **do**

  if $T_\ell + W \in \mathscr{S}_{r-\ell}$ **then**

    $\mathcal{S} \leftarrow \mathcal{S} \cup \{T_\ell + W\}$

  end if

end for

return $\mathcal{S}^{\mathrm{Stab}(T)}$

Complexity: $\#\mathscr{S}_{r-\ell} / \#\,\mathrm{Stab}(T_\ell) \approx \binom{\#\mathcal{G}}{r-\ell} \Big/ \#\,\mathrm{Stab}(T_\ell)$.

For $\ell = 3$,

$$\#\mathscr{S}_2 / \,\mathrm{Stab}(T_\ell) = 68, \;\#\,\mathrm{Stab}(T_\ell) = 32 \text{ and } \binom{49}{2} / 32 \approx 37.$$

The rank is a distance

Rank is a distance $D : (\Phi, \Psi) \mapsto \mathrm{rk}(\Phi - \Psi)$.

Example

$$D\left(\begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix}, \begin{pmatrix}0 & 0\\ 0 & 1\end{pmatrix}\right) = \mathrm{rk}\left(\begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}\right) = 2.$$

We can extend the distance D to any set of formulae.

Definition (Neighbourhood)

Let $\mathcal{S}$ and $\mathcal{S}'$ be sets of matrices. We define $D(\mathcal{S}, \mathcal{S}')$ as

$$D(\mathcal{S}, \mathcal{S}') = \min_{\Phi \in \mathcal{S}, \Psi \in \mathcal{S}'} \left(\mathrm{rk}(\Phi - \Psi)\right).$$

We denote by $\mathcal{V}_d(\mathcal{S})$ the d-neighbourhood of $\mathcal{S}$.

Theorem

Let $W \in \mathscr{S}_{r-\ell}$ *be such that*

$$T_\ell \oplus W \in \mathscr{S}_r.$$

Then there exists $\sigma \in \mathrm{Stab}(T_\ell)$ *such that*

$$W \circ \sigma \in \mathcal{V}_1(\pi_{\ell-1}) \cap \mathcal{V}_1(\pi_{\ell-2})$$

or

$$W \circ \sigma \in \mathcal{V}_1(\pi_{\ell-1}) \cap \mathcal{V}_1(\pi_{\ell-1} - \pi_{\ell-2}).$$

Example with $\ell = 3$ on $\mathbb{F}_2$

We have

$$T_3 = \mathrm{Span}(\underbrace{a_0 b_0}_{\pi_0}, \underbrace{a_1 b_0 + a_0 b_1}_{\pi_1}, \underbrace{a_2 b_0 + a_1 b_1 + a_0 b_2}_{\pi_2})$$

and the set of generators $\mathcal{G}$ satisfies $\#\mathcal{G} = 49$.

set enumerated	cardinality
$\mathscr{S}_2$	980
$\mathscr{S}_2 \cap (\mathcal{V}_1(\pi_2) \cap \mathcal{V}_1(\pi_1)) \circ \mathrm{Stab}(T_3)$	64
$\mathscr{S}_2 \cap (\mathcal{V}_1(\pi_2) \cap \mathcal{V}_1(\pi_2 - \pi_1)) \circ \mathrm{Stab}(T_3)$	144

Improved search

Improved search

Input: n, r

$\mathcal{S} \leftarrow \emptyset$

for $W \in (\mathscr{S}_{r-\ell} \cap \mathcal{V}_1(\pi_{\ell-1}) \cap \mathcal{V}_1(\pi_{\ell-2})) \,/\, \mathrm{Stab}(T_\ell)$ **do**

  if $T_\ell + W \in \mathscr{S}_r$ **then**

    $\mathcal{S} \leftarrow \mathcal{S} \cup \{T_\ell + W\}$

  end if

end for

for $W \in (\mathscr{S}_{r-\ell} \cap \mathcal{V}_1(\pi_{\ell-1}) \cap \mathcal{V}_1(\pi_{\ell-1} - \pi_{\ell-2})) \,/\, \mathrm{Stab}(T_\ell)$ **do**

  if $T_\ell + W \in \mathscr{S}_r$ **then**

    $\mathcal{S} \leftarrow \mathcal{S} \cup \{T_\ell + W\}$

  end if

end for

return $\mathcal{S}^{\mathrm{Stab}(T_\ell)}$

We compare our approach to the search with stabilizer:

product	time (s)	est. speed-up	nb. of solutions
ShortProduct_4	3.0	10	$1,440$
ShortProduct_5	$2.4 \cdot 10^3$	10^5	$146,944$

Table: Computation of decompositions of the short product on a single core 3.3 GHz Intel Core i5-4590.

Matrix product 3×2 by 2×3

$\Pi_{p,q,r}$: the bilinear map
$\pi_{i,j}$: the bilinear forms of the coefficients

Equations for $\Pi_{3,2,3}$:

- $\mathscr{S}_6 \cap \mathcal{V}_1(\pi_{1,1} + \pi_{2,2} + \pi_{3,3})$
- $\mathscr{S}_6 \cap \mathcal{V}_1(\pi_{1,1} + \pi_{2,2}) \cap \mathcal{V}_1(\pi_{1,1} + \pi_{3,3})$
- $\mathscr{S}_6 \cap \mathcal{V}_1(\pi_{1,1} + \pi_{2,2}) \cap \mathcal{V}_1(\pi_{1,2} + \pi_{3,3})$
- $\mathscr{S}_6 \cap \mathcal{V}_1(\pi_{1,1} + \pi_{2,2}) \cap \mathcal{V}_1(\pi_{3,3})$
- $\mathscr{S}_6 \cap \mathcal{V}_1(\pi_{1,1}) \cap \mathcal{V}_1(\pi_{2,2}) \cap \mathcal{V}_1(\pi_{3,3})$

product	time (s)	est. speed-up	nb. of solutions
2×3 by 3×2	$4.1 \cdot 10^6$	10^9	$1,096,452$
3×2 by 2×3	$3.0 \cdot 10^6$	10^4	$7,056$

Table: Computation of decompositions of the matrix product on a single core 3.3 GHz Intel Core i5-4590.

Conclusion

We obtain interesting speed-up for symmetric bilinear maps such as matrix product and short product.

What kind of predicates for polynomials product (small group of symmetry)?

How to push computations further: possible to decompose matrix product 3×3 by 3×3?